

glimpathcr: An R Package for Ordinal Response Prediction in High-dimensional Data Settings

Kellie J. Archer

Virginia Commonwealth University

Abstract

This paper describes an R package, **glimpathcr**, that provides a function for fitting a penalized continuation ratio model when interest lies in predicting an ordinal response. The function, `glimpath.cr` uses the coordinate descent fitting algorithm as implemented in `glimpath` and described by (Park and Hastie 2007a). Methods for extracting all estimated coefficients, extracting non-zero coefficient estimates, obtaining the predicted class, and obtaining the class-specific fitted probabilities have been implemented. Additionally, generic methods from `glimpath` including `summary`, `print`, and `plot` can be applied to a `glimpath.cr` object.

Keywords: ordinal response, penalized models, LASSO, L₁ constraint, R.

1. Introduction

High-throughput genomic experiments are frequently conducted for the purpose of examining whether genes are predictive of or significantly associated with phenotype. In many biomedical settings where histopathological or health status data are collected, phenotypic variables are recorded on an ordinal scale. Nevertheless, most often investigators neglect the ordinality of the phenotypic data and rather dichotomize the ordinal class than apply statistical methods suitable for two-class comparisons and predictions. This tendency to analyze ordinal data using dichotomous class methodologies may be due to the lack of available statistical methods and software for modeling an ordinal response in the presence of a high-dimensional covariate space. The approach of collapsing ordinal categories may neglect important information in the study (Armstrong and Sloan 1989).

A variety of statistical modeling procedures, namely, proportional odds, adjacent category, stereotype logit, and continuation ratio models can be used to predict an ordinal response. In this paper, we focus attention to the continuation ratio model because its likelihood can be easily re-expressed such that existing software can be readily adapted and used for model fitting. Suppose for each observation, $i = 1, \dots, n$, the response Y_i belongs to one ordinal class $k = 1, \dots, K$ and \mathbf{x}_i represents a p -length vector of covariates. The backward formulation of the continuation ratio models the logit as

$$\text{logit}(P(Y = k | Y \leq k, \mathbf{X} = \mathbf{x})) = \alpha_k + \boldsymbol{\beta}_k^T \mathbf{x} \quad (1)$$

whereas the forward formulation models the logit as

$$\text{logit}(P(Y = k | Y \geq k, \mathbf{X} = \mathbf{x})) = \alpha_k + \boldsymbol{\beta}_k^T \mathbf{x}. \quad (2)$$

Rather than describe both formulations in detail, here we present the backward formulation, which is commonly used when progression through disease states from none, mild, moderate, severe is represented by increasing integer values, and interest lies in estimating the odds of more severe disease compared to less severe disease (Bender and Benner 2000). Therefore for $i = 1, \dots, n$ we can construct a vector \mathbf{y}_i from Y_i to represent ordinal class membership, such that $\mathbf{y}_i = (y_{i1}, y_{i2}, \dots, y_{iK})^T$, where $y_{ik} = 1$ if the response is in category k and 0 otherwise, so that $n_i = \sum_{k=1}^K y_{ik} = 1$. Using the logit link, the equation representing the conditional probability for class k is

$$\delta_k(\mathbf{x}) = P(Y = k | Y \leq k, \mathbf{X} = \mathbf{x}) = \frac{\exp(\alpha_k + \boldsymbol{\beta}^T \mathbf{x})}{1 + \exp(\alpha_k + \boldsymbol{\beta}^T \mathbf{x})}. \quad (3)$$

The likelihood for the continuation ratio model is then the product of conditionally independent binomial terms (Cox 1975), which is given by

$$L(\boldsymbol{\beta} | \mathbf{y}, \mathbf{x}) = \prod_{i=1}^n \delta_2^{y_{i2}} (1 - \delta_2)^{1 - \sum_{k=2}^K y_{ik}} \times \dots \times \delta_K^{y_{iK}} (1 - \delta_K)^{1 - y_{iK}} \quad (4)$$

where here we have simplified our notation by not explicitly including the dependence of the conditional probability δ_k on \mathbf{x} . Further, simplifying our notation to let $\boldsymbol{\beta}$ represent the vector containing both the thresholds $(\alpha_2, \dots, \alpha_K)$ and the log odds $(\beta_1, \dots, \beta_p)$ for all $K - 1$ logits, the full parameter vector is

$$\boldsymbol{\beta} = (\alpha_2, \beta_{21}, \beta_{22}, \dots, \beta_{2p}, \dots, \alpha_K, \beta_{K,1}, \beta_{K,2}, \dots, \beta_{K,p})^T \quad (5)$$

which is of length $(K - 1)(p + 1)$. As can be seen from equation 4, the likelihood can be factored into $K - 1$ independent likelihoods, so that maximization of the independent likelihoods will lead to an overall maximum likelihood estimate for all terms in the model (Bender and Benner 2000). A model consisting of $K - 1$ different $\boldsymbol{\beta}$ vectors may be overparameterized so to simplify, one commonly fits a constrained continuation model, which includes the $K - 1$ thresholds $(\alpha_2, \dots, \alpha_K)$ and one common set of p slope parameters, $(\beta_1, \dots, \beta_p)$. To fit a constrained continuation ratio model, the original dataset can be restructured by forming $K - 1$ subsets, where for classes $k = 2, \dots, K$, the subset contains those observations in the original dataset up to class k . Additionally, for the k^{th} subset, the outcome is dichotomized as $y = 1$ if the ordinal class is k and $y = 0$ otherwise. Furthermore, an indicator is constructed for each subset representing subset membership. Thereafter the $K - 1$ subsets are appended to form the restructured dataset, which represents the $K - 1$ conditionally independent datasets in equation 4. Applying a logistic regression model to this restructured dataset yields an L_1 constrained continuation ratio model.

2. Penalized Models

For datasets where the number of covariates p exceeds the sample size n , the backwards stepwise procedure cannot be undertaken. Furthermore, for any problem using a forward selection procedure the discrete variable inclusion process can exhibit high variance. Moreover, for high-dimensional covariate spaces, the best subset procedure is computationally prohibitive. Two penalized methods, ridge and L_1 penalization, places a penalty on a function of the coefficient

estimates, thereby permitting a model fit even for high-dimensional data Tibshirani (1996, 1997). A generalization of these penalized models can be expressed as,

$$\tilde{\beta} = \arg \min_{\beta} \left(\sum_{i=1}^n (y_i - \beta_0 - \sum_{j=1}^p x_{ij} \beta_j)^2 + \lambda \sum_{j=1}^p |\beta_j|^q \right) \quad (6)$$

for $q \geq 0$. When $q = 1$ we have the an L_1 penalized model, when $q = 2$ we have ridge regression. Values of $q \in (1, 2)$ provide a compromise between the L_1 and ridge penalized models. Because when $q > 1$ coefficients are no longer set exactly equal to 0, the elastic net penalty was introduced

$$\lambda \sum_{j=1}^p (\alpha \beta_j^2 + (1 - \alpha) |\beta_j|). \quad (7)$$

3. Implementation

The **glmptcr** package was written in the R programming environment (R Development Core Team 2009) and depends on the **glmptcr** package (Park and Hastie 2007b). Similar to the **Design** package which includes a function `cr.setup` for restructuring a dataset for fitting a forward continuation ratio model, in this package the model is fit by restructuring the dataset then passing the restructured dataset to a penalized logistic regression fitting function. However, unlike `cr.setup` which produces an object of class `list` from which the response and restructured independent variables are extracted and passed to a model fitting algorithm, in the **glmptcr** package the restructuring functions are transparent to the user. Specifically, the **glmptcr** package fits either a forward or backward (default) penalized constrained continuation ratio model by specification of `method="forward"` in the `glmptcr.cr` call. The `glmptcr.cr` function restructures the dataset to represent the $K - 1$ conditionally independent likelihoods and then fits the penalized continuation ratio model using the **glmptcr** framework. Therefore, the predictor-corrector fitting procedure used by the `glmptcr` function in the **glmptcr** package is used in fitting the penalized continuation ratio model when invoking `glmptcr.cr`. This allows fitting a penalized model for situations where the number of covariates p exceed the sample size n . In addition, methods for extracting the best fitting model from the path using AIC and BIC criteria, obtaining predicted class and fitted class probabilities, and returning coefficient estimates were written in addition to adapting the `print`, `summary`, and `plot` methods from **glmptcr** for a `glmptcr.cr` object.

4. Example

The **glmptcr** package includes a filtered microarray dataset `diabetes` in which asymptomatic males not previously diagnosed with Type II diabetes were enrolled and subsequently were cross-classified as either normal controls (N=8), having impaired fasting glucose (N=7), or as Type II diabetics (N=9) based on a fasting glucose intolerance test. From the code below we can see that the classification variable is stored as `y` in the first column of the `diabetes.data.frame`; all subsequent columns are the 11,066 Illumina probes having no negative expression values. In fitting the model we can extract the covariates into an object `x` and the

ordinal outcome into the object `y`. The code for fitting a backward (default) continuation ratio model is given by

```
> library(glmpathcr)
> data(diabetes)
> dim(diabetes)
```

```
[1] 24 11067
```

```
> names(diabetes)[1:10]
```

```
[1] "y" "ILMN_1343291" "ILMN_1651228" "ILMN_1651229" "ILMN_1651236" "ILMN_1651237"
[8] "ILMN_1651268" "ILMN_1651278" "ILMN_1651286"
```

```
> summary(diabetes$y)
```

```
control impaired fasting glucose type 2 diabetes
      8              7              9
```

```
> x <- diabetes[, 2:dim(diabetes)[2]]
> y <- diabetes$y
> fit <- glmpath.cr(x, y)
```

As with `glmpath` model objects, methods such as `summary` and `plot` can be applied to `glmpath.cr` model objects, which are helpful for selecting the step at which to select the final model from the solution path.

```
> summary(fit)
```

	Df	Deviance	AIC	BIC
Step 1	3	5.248273e+01	58.48273	62.01690
Step 2	4	2.334393e+01	33.34393	39.23419
Step 5	5	2.047337e+01	30.47337	36.36364
Step 7	6	2.013650e+01	32.13650	39.20482
Step 10	7	1.587273e+01	29.87273	38.11911
Step 14	8	7.939954e+00	23.93995	33.36438
Step 17	9	7.530910e+00	25.53091	36.13339
Step 19	10	7.327910e+00	27.32791	39.10845
Step 21	11	7.055827e+00	29.05583	42.01442
Step 24	12	6.608876e+00	30.60888	44.74552
Step 28	13	3.583362e+00	29.58336	44.89806
Step 31	14	3.176651e+00	31.17665	47.66940
Step 34	15	2.491539e+00	32.49154	50.16235
Step 38	16	1.803196e+00	33.80320	52.65206
Step 41	17	1.718769e+00	35.71877	55.74568
Step 46	18	4.362179e-01	36.43622	57.64119

Step 48	18	9.655223e-02	36.09655	57.30152
Step 53	18	1.378812e-02	36.01379	57.21876
Step 56	19	1.012908e-02	40.01013	63.57121
Step 57	19	6.701787e-03	40.00670	63.56778
Step 58	20	6.638610e-03	40.00664	63.56772
Step 60	21	4.759464e-03	42.00476	66.74389
Step 61	22	4.746349e-03	44.00475	69.92193
Step 63	23	4.429073e-03	46.00443	73.09967
Step 65	24	4.097855e-03	48.00410	76.27739
Step 66	24	3.732952e-03	48.00373	76.27702
Step 67	23	3.581300e-03	46.00358	73.09882
Step 68	22	3.121120e-03	44.00312	69.92031
Step 69	21	3.063817e-03	42.00306	66.74219
Step 70	20	3.021383e-03	40.00302	63.56410
Step 71	19	2.718212e-03	38.00272	60.38574
Step 72	18	2.469716e-03	36.00247	57.20744
Step 73	17	2.349066e-03	34.00235	54.02926
Step 75	17	2.207586e-03	34.00221	54.02912
Step 76	18	2.104317e-03	36.00210	57.20707
Step 77	19	2.089189e-03	38.00209	60.38511
Step 78	20	2.057689e-03	40.00206	63.56313
Step 80	21	1.984377e-03	42.00198	66.74111
Step 82	22	1.952013e-03	44.00195	69.91914
Step 83	22	1.943365e-03	44.00194	69.91913
Step 85	22	1.665566e-03	46.00167	73.09690
Step 87	23	1.475808e-03	46.00148	73.09671
Step 89	24	1.419080e-03	48.00142	76.27471
Step 91	25	1.392577e-03	50.00139	79.45274
Step 92	25	1.372547e-03	50.00137	79.45272
Step 93	25	1.318691e-03	50.00132	79.45266
Step 95	26	1.248295e-03	52.00125	82.63065
Step 96	27	1.221342e-03	54.00122	85.80867
Step 97	28	1.167589e-03	58.00117	92.16473
Step 98	28	1.161724e-03	56.00116	88.98667
Step 99	27	1.135028e-03	54.00114	85.80859
Step 100	27	1.130176e-03	54.00113	85.80858
Step 101	27	1.121360e-03	54.00112	85.80857
Step 103	27	1.107591e-03	54.00111	85.80856
Step 104	28	1.102221e-03	56.00110	88.98661
Step 106	29	1.043946e-03	58.00104	92.16461
Step 107	30	1.009175e-03	60.00101	95.34262
Step 108	31	9.025019e-04	62.00090	98.52057
Step 109	32	8.991575e-04	64.00090	101.69862
Step 111	33	8.588050e-04	66.00086	104.87664
Step 112	34	8.523746e-04	68.00085	108.05468
Step 113	35	8.409853e-04	70.00084	111.23273
Step 114	35	8.396445e-04	70.00084	111.23272

```

Step 115 35 7.788606e-04 70.00078 111.23266
Step 116 36 7.567292e-04 72.00076 114.41069
Step 118 37 6.879621e-04 74.00069 117.58868
Step 119 38 6.848739e-04 76.00068 120.76673

```

```
> plot(fit, xvar = "step", type = "bic")
```

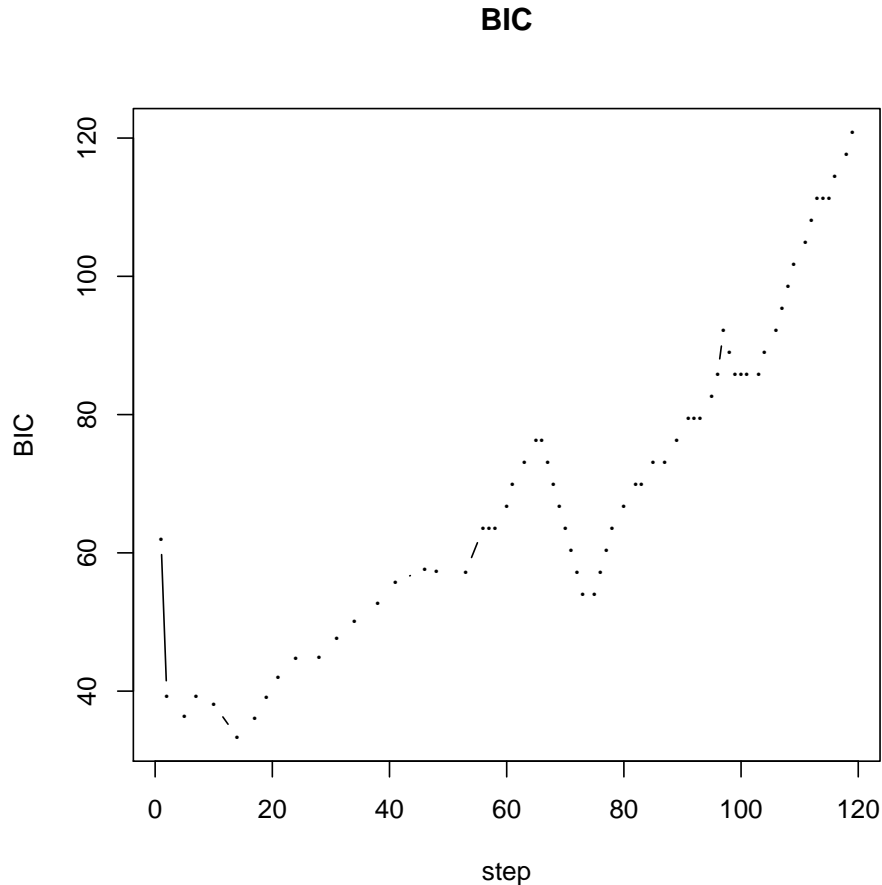


Figure 1: Plot of regularization path for `glmpath.cr` object using simulated dataset, `data`.

Note that when plotting, the horizontal axis can be `norm`, `lambda`, or `step`, however extractor functions for `glmpath.cr` generally require the step to be selected, so we have selected `xvar = "step"` in this example. The vertical axis can be coefficients, `aic` or `bic`. As one can see, there is a multitude of models fit from one call to `glmpath.cr`. To facilitate extraction of best fitting models using commonly used criterion, the `model.select` function can be used. The `model.select` function extracts the best fitting model from the solution path, where the `which` parameter allows one to select either AIC or by default, BIC.

```

> BIC.step <- model.select(fit)
> BIC.step

```

```
[1] 14
```

```
> AIC.step <- model.select(fit)
> AIC.step
```

```
[1] 14
```

In this example, the minimum BIC corresponds to a 8 degree of freedom model.

The `coef` function returns all estimated coefficients for a `glmpath.cr` fitted model, where the model selected is indicated by step number, `s`. The `nonzero.coef` function returns only those non-zero coefficient estimates for a selected model.

```
> coefficients <- coef(fit, s = BIC.step)
> sum(coefficients != 0)
```

```
[1] 8
```

```
> nonzero.coef(fit, s = BIC.step)
```

```
Intercept  ILMN_1705116  ILMN_1733757  ILMN_1758311  ILMN_1759232  ILMN_2100437
18.3204862012  0.0334955982 -0.0003751318  0.0022607450 -0.0347775941 -0.0001679020  2.016
      cp2
-2.0164341480
```

Note that the `glmpath.cr` function fits a penalized constrained continuation ratio model; therefore for K classes, there will be $K - 1$ intercepts representing the cutpoints between adjacent classes. In this package, the nomenclature for these cutpoints is to use “cp k ” where $k = 1, \dots, K - 1$. In this dataset, $K = 3$ so the intercepts are `cp1` and `cp2` with `Intercept` being an offset. The probe having the largest absolute coefficient estimate, `ILMN_1759232`, corresponds to the insulin receptor substrate 1 (IRS1) gene which is biologically meaningful.

Continuation ratio models predicts conditional probabilities so a new method to extract the fitted probabilities and predicted class was created. The `predict` and `fitted` functions are equivalent, and return either the predicted class or the fitted probabilities from the penalized continuation ratio model for a `glmpath.cr` object. The user is required to supply the fitted `glmpath.cr` model object, a data matrix `newx` that is either the same as the training data or an independent dataset having the same number and order of covariates as the training data, a vector `newy` that provides the class labels of the ordinal response. These functions extract the fitted values for the best fitting model using the BIC criteria by default, which can be changed to extracting the best fitting AIC model by supplying `which="AIC"`. By default, the predicted class is output. If one desired the fitted class-specific probabilities from the model, the `type="probs"` argument should be supplied.

```
> pred <- predict(fit)
> table(pred, y)
```

```

              y
pred          control impaired fasting glucose type 2 diabetes
control              8                0                0
impaired fasting glucose 0                7                0
type 2 diabetes         0                0                9

```

```

> pred <- predict(fit, type = "probs")
> pred

```

```

      control impaired fasting glucose type 2 diabetes
[1,] 8.178419e-01      0.17830887      0.0038492509
[2,] 9.239276e-01      0.07464256      0.0014297981
[3,] 8.947141e-01      0.10324488      0.0020410068
[4,] 9.530268e-01      0.04611633      0.0008568896
[5,] 8.835483e-01      0.11416679      0.0022848811
[6,] 9.070107e-01      0.09121017      0.0017791192
[7,] 8.672968e-01      0.13005260      0.0026506120
[8,] 8.784051e-01      0.11919565      0.0023992189
[9,] 7.140319e-02      0.76841557      0.1601812424
[10,] 2.703789e-01      0.68655530      0.0430657815
[11,] 5.059698e-02      0.74290859      0.2064944307
[12,] 9.145857e-02      0.77758826      0.1309531683
[13,] 5.568355e-02      0.75132840      0.1929880460
[14,] 8.375911e-02      0.77530692      0.1409339749
[15,] 1.423227e-01      0.77016247      0.0875147915
[16,] 3.278482e-04      0.12689758      0.8727745671
[17,] 4.987756e-04      0.15400425      0.8454969779
[18,] 1.249965e-04      0.08046247      0.9194125346
[19,] 1.588154e-04      0.09020377      0.9096374103
[20,] 1.239401e-04      0.08013625      0.9197398137
[21,] 1.891746e-04      0.09801512      0.9017957098
[22,] 8.823833e-04      0.19912988      0.7999877345
[23,] 6.318904e-05      0.05792144      0.9420153757
[24,] 1.585017e-04      0.09011893      0.9097225667

```

For illustrative purposes, a forward continuation ratio model can be fit using the syntax

```

> fit <- glmpath.cr(x, y, method = "forward")

```

As before, the parameter estimates corresponding to the model attaining the minimum BIC can be extracted using the following code.

```

> coefficients <- coef(fit, s = BIC.step)
> nonzero.coef(fit, s = BIC.step)

```

```

      Intercept  ILMN_1701911  ILMN_1705116  ILMN_1733757  ILMN_1759232  ILMN_2100437
-8.878742e+00  6.455187e-03  -2.238348e-02  2.963308e-04  2.964169e-02  3.320597e-05  -1.601
      cp2
1.601889e+00

```


and the predicted class can be obtained using

```
> pred <- predict(fit)
> table(pred, y)
```

pred	y		
	control	impaired fasting glucose	type 2 diabetes
control	8	1	0
impaired fasting glucose	0	6	0
type 2 diabetes	0	0	9

Summary

Herein we have described the **glmptcr** package which works in conjunction with the **glmptcr** package in the R programming environment. The package provides methods for fitting either a forward or backward penalized continuation ratio model. Moreover, the likelihood-based penalized continuation ratios models have been demonstrated to have good performance when applied to microarray gene expression datasets (Archer and Williams 2011) in comparison to corresponding penalized Bayesian continuation ratio models (Kiiveri 2008). A similar package, **glmnetcr**, which uses the **glmnet** fitting algorithm for fitting a penalized constrained continuation ratio model has also been developed and is available for download from the Comprehensive R Archive Network. Functions for extracting coefficients, extracting non-zero coefficients, and obtaining fitted probabilities and predicted class in the **glmnetcr** package follow those in **glmptcr** and both packages have similar performance (Archer and Williams 2011). Therefore either the **glmptcr** or **glmnetcr** package should be helpful when predicting an ordinal response for datasets where the number of covariates exceeds the number of available samples.

Acknowledgments

This research was supported by the National Institute of Library Medicine R03LM009347.

References

- Archer KJ, Williams AA (2011). “L₁ penalized continuation ratio models for ordinal response prediction using high-dimensional datasets.” *Statistics in Medicine*, **revision under review**.
- Armstrong B, Sloan M (1989). “Ordinal regression models for epidemiologic data.” *American Journal of Epidemiology*, **129**, 191–204.
- Bender R, Benner A (2000). “Calculating ordinal regression models in SAS and S-Plus.” *Biometrical Journal*, **42**, 677–699.
- Cox D (1975). “Partial likelihood.” *Biometrika*, **62**, 269–276.

- Kiiveri HT (2008). “A general approach to simultaneous model fitting and variable elimination in response models for biological data with many more variables than observations.” *BMC Bioinformatics*, **9**, 195.
- Park M, Hastie T (2007a). “L1-regularization path algorithm for generalized linear models.” *Journal of the Royal Statistical Society, B*, **64**, 659–677.
- Park MY, Hastie T (2007b). *glmPath: L1 Regularization Path for Generalized Linear Models and Cox Proportional Hazards Model*. R package version 0.94.
- R Development Core Team (2009). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. ISBN 3-900051-07-0, URL <http://www.R-project.org>.
- Tibshirani R (1996). “Regression shrinkage and selection via the Lasso.” *Journal of the Royal Statistical Society, B*, **58**, 267–288.
- Tibshirani R (1997). “The lasso method for variable selection in the Cox model.” *Statistics in Medicine*, **16**, 385–395.

Affiliation:

Kellie J. Archer
Department of Biostatistics
Virginia Commonwealth University
Box 980032
Richmond, VA 23298-0032
E-mail: kjarcher@vcu.edu
URL: <http://www.people.vcu.edu/~kjarcher/>